

Docket Number RA-5548
Examiner Russell Guill, GUA 2123

Office Action Response
June 11, 2007

RECEIVED
CENTRAL FAX CENTER

JUN 11 2007

Please Amend the Claims as Follows:

1 1. (Original) A computer system supported method for preparing a test source
2 file for verifying the performance of a simulated cache memory integrated circuit
3 device design, comprising the steps of:
4 sequentially and randomly creating a series of functions;
5 updating a data integrity buffer after each function of said series of
6 functions is created;
7 creating a series of integrity check functions from said data integrity buffer;
8 and
9 writing said series of functions and said series of integrity check functions
10 to a test file.

1 2. (Original) The method of claim 1, wherein the data integrity buffer includes a
2 plurality of records, each record of said plurality of records including a cache
3 memory address associated with contents of that cache memory address.

1 3. (Original) The method of claim 2, wherein an address of said each record and
2 contents of said each record are generated at random.

1 4. (Original) The method of claim 3, wherein the data integrity buffer is updated
2 after a normal function is created in said series of functions.

Docket Number RA-5548
Examiner Russell Guill, GUA 2123

Office Action Response
June 11, 2007

1 5. (Original) The method of claim 4, wherein at least one of the series of
2 functions includes a bitwise memory write operation to a partial word cache
3 memory location and wherein a corresponding integrity check function includes a
4 bitwise memory read operation to read said partial word cache memory location.

1 6. (Original) The method of claim 2, further comprising the steps of reading an
2 entire test settings file into an input buffer before the step of sequentially and
3 randomly creating a series of functions, and further wherein the step of
4 sequentially and randomly creating includes the step of using parameter data
5 from the test settings file to create said series of functions.

1 7. (Original) The method of claim 6, wherein the test settings file includes data
2 directing the generation of prefetch loops.

1 8. (Original) The method of claim 3, wherein the address of each record and the
2 contents of each record are generated at random using the Mitchell-Moore
3 Additive generation method.

1 9. (Currently Amended) A digital storage medium having stored thereon a
2 sequence of digital instructions executable by a computer to configure ~~a~~the
3 computer to prepare a test source file for use in verifying the contents of a
4 simulated cache memory integrated circuit device design, the digital instructions
5 defining a sequence of steps for:

6 sequentially and randomly creating a series of functions;

Docket Number RA-5548
Examiner Russell Guill, GUA 2123

Office Action Response
June 11, 2007

7 updating a data integrity buffer in said computer after each function of said
8 series of functions is created;
9 creating a series of integrity check functions from said data integrity buffer;
10 and
11 writing said series of functions and said series of integrity check functions
12 to a test file.

1 10. (Original) The method of claim 9, wherein the data integrity buffer includes a
2 plurality of records, each record including a cache memory address associated
3 with contents of that cache memory address.

1 11. (Original) The method of claim 10, wherein an address of each record and
2 contents of each record are generated at random.

1 12. (Original) The method of claim 11, wherein the data integrity buffer is updated
2 after a normal function is created in said series of functions.

1 13. (Original) The method of claim 12, wherein at least one of the series of
2 functions includes a bitwise memory write operation to a partial word cache
3 memory location and wherein a corresponding integrity check function includes a
4 bitwise memory read operation to read said partial word cache memory location.

1 14. (Original) The method of claim 10, further comprising the steps of reading an
2 entire test settings file into an input buffer before the step of sequentially and

Docket Number RA-5548
Examiner Russell Guill, GUA 2123

Office Action Response
June 11, 2007

- 3 randomly creating a series of functions, and further wherein the step of
4 sequentially and randomly creating includes the step of using parameter data
5 from the test settings file to create said series of functions.

- 1 15. (Original) The method of claim 14, wherein the test settings file includes data
2 directing the generation of prefetch loops.

- 1 16. (Original) The method of claim 11, wherein an address of each record and
2 contents of each record are generated at random using the Mitchell-Moore
3 Additive generation method.

- 1 17. (Original) A digital computer configured to prepare a test source file for
2 verifying the performance of a simulated cache memory integrated circuit device
3 design, said computer including:

4 means for sequentially creating a series of functions;

5 means for updating a data integrity buffer in said computer after each
6 function of said series of functions is created;

7 means for creating a series of integrity check functions from data in said
8 data integrity buffer; and

9 means for writing said series of functions and said series of integrity check
10 functions to a test file.

- 1 18. (Original) The computer of claim 17 further comprising a means for providing
2 test settings to said means for sequentially creating the series of instructions.

Docket Number RA-5548
Examiner Russell Guill, GUA 2123

Office Action Response
June 11, 2007

1 19. (Original) The computer of claim 18, wherein the data integrity buffer includes
2 a plurality of records, each record including a cache memory address associated
3 with contents of that cache memory address, and further wherein the digital
4 computer further includes a means for randomly generating an address of each
5 record and contents of each record are generated at random.

1 20. (Original) The computer of claim 19, further comprising means for updating
2 the data integrity buffer.

1 21. (Original) The computer of claim 20, wherein the means for updating the data
2 integrity buffer includes means for updating the data integrity buffer after a
3 normal function is created in said series of functions.

1 22. (Original) The computer of claim 21, wherein means for providing test
2 settings to said means for sequentially creating the series of instructions includes
3 means for directing the generation of prefetch loops.